

NeXT Development Tools

The rich library of development tools bundled with NeXT computers lets you get the most out of NeXTstep. At each stage of development—design, implementation, compilation, debugging, and tuning—there are support tools that help you efficiently create robust applications.

Interface Builder

As discussed in previous sections, you use Interface Builder to design a graphical interface for your application. Instead of writing code, you manipulate graphic representations of objects just as if you were using a graphics editor to create a drawing. As you design the look of your application, you also make graphical connections between objects, describing the overall structure of your program. For instance, you can create a slider and a text field, and connect them so that the text field is continuously updated as the user moves the slider's knob up and down.

Interface Builder's ability to create and control projects makes it ideal as the hub for program development. It keeps track of the files used for building an application—source files, image files, and sound files, to name a few—and even specifies dependencies for the code files so that when you rebuild your project, only the parts that have changed are recompiled.

Editors

Edit, NeXT's simple yet powerful mouse-based editor, is well suited for programming. Edit manages logical blocks of source code. For example, you can double-click a brace, bracket, or parenthesis to select the text included between it

and its matching symbol. Edit can also collapse and expand logical blocks of code so you can see the overall structure.

In addition to the services that all applications enjoy (such as looking up a class specification in the *NeXT Developer's Library*), Edit can look up sections of the UNIX manual directly, and can run its current document through a text filter for additional formatting.

In addition to Edit, NeXT computers come with GNU Emacs, an extensible terminal-based editor, and vi, the native UNIX screen editor.

Compilers

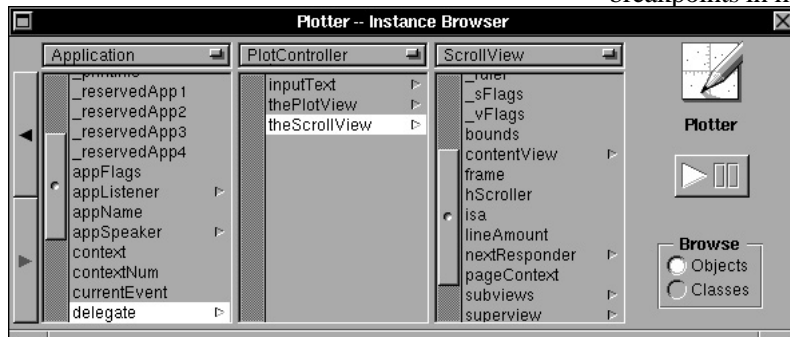
The C compiler used on NeXT computers is based on the GNU C compiler from the Free Software Foundation. It accepts ANSI C as well as more traditional C code. NeXT has extended the compiler to compile Objective-C code as well. The GNU C compiler produces highly optimized object code that is nonetheless compatible with GDB, the debugger described below.

In addition, NeXT computers come with a C++ compiler, also from the Free Software Foundation. With this compiler, you can use C++ code in conjunction with Objective-C code. This gives you complete access to the Application Kit even if your program's internal routines are implemented in C++. Unlike most C++ systems, this compiler compiles C++ source directly instead of first translating it into C.

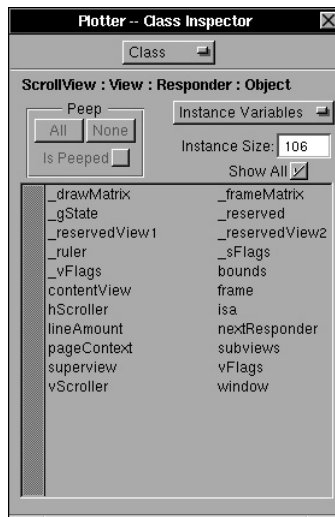
Debugging and Performance Tools

NeXT provides a host of applications and utilities to help you debug your program and improve its performance.

AppInspector



AppInspector is a tool that lets you inspect a running application, examine its objects and variables, and perform run-time tracing of Objective-C messages. It also shows information about classes, such as their methods. It can be run standalone or from the debugger. If an object's variables are standard types, you can inspect them with AppInspector—and since outlets (references to other objects) are just a kind of variable, you can examine the whole network of objects that make up an application.



GDB

GDB, the source- and assembly-level debugger from the Free Software Foundation, has been modified by NeXT to work with the Objective-C and C++ languages, as well as with constructs of the Mach operating system. With GDB's object-oriented extensions, you can set breakpoints in methods and send messages to

objects in real time. GDB is one of the few debuggers on any platform that supports source-level debugging of optimized object code.

GDB can access and display source files for you. As you step through a program, GDB can open the

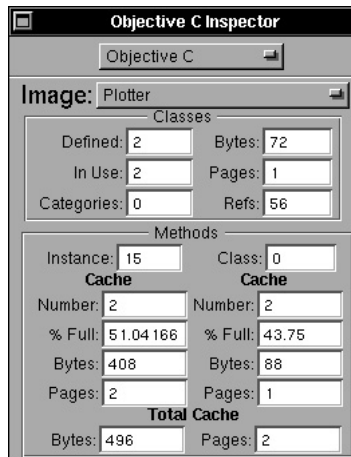
appropriate files with Edit and display the line currently executing. GDB also has features that make it easy to debug programs that weren't started with debugging in mind: GDB can debug core dumps and can attach to running processes.

MallocDebug



MallocDebug measures an application's use of dynamic memory. It can measure all allocated memory in an application or just the memory allocated since a given time. MallocDebug also detects incorrectly written memory, so you can tell if you've written data before the beginning or past the end of a block. It also contains a garbage detector that you can use to detect unused memory that can't be reclaimed.

ProcessMonitor

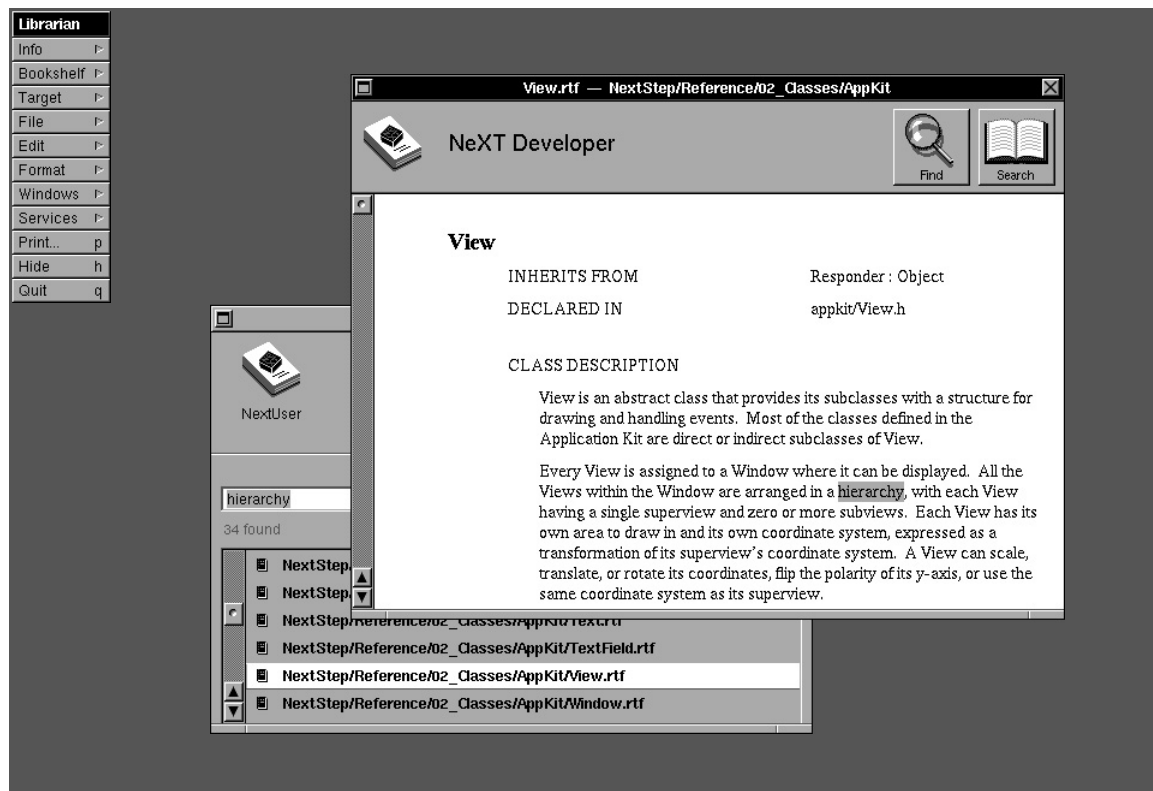


ProcessMonitor displays several types of statistics about a running process or application. You can inspect Mach, Display PostScript, and dynamic memory usage, as well as statistical information about Objective-C objects and classes. In addition, processes can be paused or killed with ProcessMonitor.

Digital Librarian

Digital Librarian is a general purpose application for finding text in any document. For programmers, it provides a convenient way to search through the *NeXT Developer's Library* or other relevant files. (The *NeXT Developer's Library* has comprehensive documentation for NeXTstep classes, C functions, and development tools.) For example, you can find all classes that have a particular method. Or, if you use Digital Librarian with your source code, you can find all occurrences of a particular variable name.

Digital Librarian finds instances of indexed words almost immediately. Since it indexes virtually every word, you can quickly find the information you're looking for. Also, Digital Librarian's services are available from any standard NeXTstep application—just select the word you want to look up and choose the Search command from the Librarian Services menu.



Tools for Development on the Digital Signal Processor (DSP)

The Digital Signal Processor (DSP), a separate processor built into NeXT computers, can be used to create, filter, and collect sounds and other real-time signals. NeXT supplies several tools for developing DSP programs:

- Motorola's cross assembler for the DSP, which creates object modules from DSP assembly source code
- BUG-56™, a DSP source-level debugger for the DSP from Ariel Corporation
- dspwrap, a program that gives a C interface to DSP assembly language macros

Tools for PostScript Development

Since all drawing on NeXT computers is done with PostScript operators (supplied either by NeXTstep objects or your own code), you need access to PostScript operators from your programs. NeXT provides this access through:

- A library routine for each PostScript operator.
- Functions you create with pswrap to execute your own PostScript routines. The pswrap program generates C functions that can execute any PostScript code. These functions can have arguments that replace variables in the PostScript code.

You also need ways to test your PostScript code. Two programs are specifically designed to give you fast turnaround when testing PostScript drawing:

- Yap, a PostScript previewer that shows PostScript code in one window and the result of its execution in another.
- pft, a shell-based program that you can use to test drawing code. pft establishes a low-level connection to the PostScript interpreter through

which you can interactively execute PostScript code and view the results on the screen.

The Mach Operating System

The Mach operating system, designed at Carnegie Mellon University, is used by all NeXT computers. Although NeXT's version of Mach is fully compatible with UNIX 4.3BSD (Berkeley Software Distribution), it was designed with different priorities, including handling multiple-processor and networked machines. For example, in each task (the equivalent of a UNIX process), Mach allows multiple concurrent threads of execution. These threads can be run on one processor or on different processors in the same machine. Mach provides low-overhead interprocess communication with a simple interface that makes it as easy to use between tasks on different computers over a network as for tasks on a single machine.

In addition to UNIX compatibility at the system call level, NeXT supplies the standard UNIX command line tools, such as awk and sed, as well as programming utilities like gprof, the UNIX profiler. You can compile and run UNIX code on a NeXT computer as easily as you can on any other computer that's based on UNIX 4.3BSD.